**Задача Eclipse**. Юний астроном Петрик сфотографував сонячне затемнення і хоче визначити, яке було затемнення – повне, часткове чи відсутнє взагалі. Він роздрукував знімок, провів  координатну пряму через центри Сонця і Місяця, визначив координати центрів зображень небесних тіл та радіуси цих зображень. Допоможіть Петрику.  Сонце і Місяць на світлині Петрика мають форму круга.

**Технічні умови**. Програма **Eclipse** читає з пристрою стандартного введення через пропуск  4 числа – координату центра та радіус Сонця, потім координату центра та радіус Місяця (всі числа натуральні, не більші 1000). Програма виводить на пристрій стандартного виведення єдине число: 0, якщо затемнення не було, 1, якщо затемнення часткове, 2,якщо повне.

**Приклади  *Введення  3 5 3 5    Виведення  2***

***Введення  3 8 3 5     Виведення  1***

*Коментар: Повне затемнення – це коли зображення диска Сонця повністю перекрите зображенням диска Місяця, часткове – це коли зображення мають більше однієї спільної точки, але перекриваються не повністю, затемнення немає, коли зображення не перекриваються зовсім, але, можливо, мають одну спільну точку.*

**Задача laying**  Бізнесмен Петро Петрович збирає речі у відпустку. З собою в літак він може взяти ручну поклажу і багаж. Для ручної поклажі у нього є рюкзак, а для багажу – здоровенна валіза.  За правилами перевезення маса ручної поклажі не повинна перевищувати ***S*** кг, а багаж може бути будь-якої маси (за наднормативний багаж Петро Петрович  готовий доплатити). Зрозуміло, найбільш цінні речі - ноутбук, фотоапарат, документи і т. д. – Петро Петрович  хоче покласти в ручну поклажу. Петро Петрович  розклав усі свої речі в порядку зменшення їх цінності і починає складати найбільш цінні речі в рюкзак. Він діє в такий спосіб - бере найцінніший предмет, і якщо його маса не перевищує ***S***, то кладе його в рюкзак, інакше кладе його до валізи. Потім він бере наступний за цінністю предмет, якщо його можна покласти в рюкзак, тобто якщо його маса разом з масою вже покладених в рюкзак речей не перевищує ***S***, то кладе його в рюкзак, інакше до валізи, і таким же чином процес триває для всіх предметів в порядку спадання їх цінності. Визначте вагу рюкзака і валізи після того, як Петро Петрович складе всі речі.

**Технічні умови**. Програма **laying** читає з пристрою стандартного введення число ***S (1 ≤ S ≤ 2 × 109)*** – максимально дозволенa вага рюкзака та число ***N (1 ≤ N ≤ 105)*** - кількість предметів.  У наступному  рядку дано ***N***чисел через пропуски***-*** маси предметів, самі предмети перераховані в порядку спадання цінності (спочатку вказана маса найціннішого предмета, потім маса другого по цінності предмета і т. д.). Всі числа натуральні, сума ваг всіх предметів не перевищує 2 × 109. Програма виводить на пристрій стандартного виведення два числа - вагу рюкзака і вагу валізи (вага порожнього рюкзака і валізи не враховується).

***Приклад***

|  |  |
| --- | --- |
| **Введення** | **Виведення** |
| 10 4  6 3 2 1 | 10 2 |

**Задача Bear**  Маленьке ведмежа йде по дорозi, вздовж якої на вiдстанi М одне вiд одного ростуть дерева. Зупиняючись пiд кожним деревом, ведмежа забуває, звiдки прийшло, i, рушаючи через деякий час в подальші мандри, випадково вибирає той чи iнший напрямок руху. На якiй вiдстанi вiд "стартового" дерева може бути ведмежа пiсля К етапiв?

**Технічні умови**. Програма **Bear** зчитує з клавiатури числа **M** та **К** через пропуск (1<=М,К<=10000). Програма виводить на екран в один рядок через пропуски вiдстанi, на яких може знаходитись ведмедик (вiд меншої до бiльшої).

***Приклад   Введення  2 6   Виведення  0 4 8 12***

**Задача Chocolates**. Петрик святкував день народження 3 листопада і  вирішив пригостити однокласників шоколадками. Шоколадка коштувала ***N*** грн. З першого листопада вартість шоколадки збільшилась рівно на ***Р*** відсотків. Визначте скільки шоколадок зможе купити Петрик на ***S*** грн після подорожчання.

**Технічні умови**. Програма **Chocolates**  читає з пристрою стандартного введення (клавіатури) 3 цілих числа: ***N (1 ≤ N ≤ 107)*** - вартість шоколадки до подорожчання,  ***Р (0 ≤ Р ≤ 100)*** - величина подорожчання шоколадки у відсотках,  ***S (1 ≤ S ≤ 107)*** - сума грошей, яка є у Петрика. Програма виводить одне число - кількість шоколадок, які може купити Петрик.

***Приклад***

|  |  |
| --- | --- |
| **Введення** | **Виведення** |
| 25 5 100 | 3 |

**Задача Mountain.**Для поповнення бюджету  країни, що відома   своїми гірськими  маршрутами, ввели новий податок для альпіністів. Величина податку пропорційна довжині маршруту, але, оскільки маршрут проходить по горах і пройдену відстань, яка залежить від висоти спуску і підйому, підрахувати складно, податок утримується без урахування висоти, тобто величина податку пропорційна горизонтальному переміщенню туристичної групи. Крім того, в силу старовинного звичаю усі туристичні групи повинні переміщатися по горах  строго із заходу на схід. Експедиція хоче заощадити на податку, тому вона  розробляє гірський маршрут з мінімальною величиною податку. При цьому, оскільки маршрут є гірським, він повинен містити підйом в гору і спуск з гори, тобто на маршруті має бути точка, яка знаходиться строго вище початку і кінця маршруту.

Альпіністи склала карту, що містить інформацію про висоту гір при пересуванні із заходу на схід. Висоти гір виміряні в точках через рівні відстані. Знайдіть на цій карті туристичний маршрут, за який податок буде мінімальний, а підйом та спуск, як і личить альпіністам, буде на маршруті.

**Технічні умови.**Програма **Mountain** читає з пристрою стандартного введення  число ***N*** - кількість точок на карті гір. У наступному рядку ***N*** чисел через пропуск містять інформацію про висоту гір в даних ***N*** точках при русі із заходу на схід. Всі числа натуральні, не перевищують **105**. Програма виводить два числа - номер точки початку маршруту і номер точки закінчення маршруту. Точки нумеруються від **1** до ***N***. Якщо маршруту, що задовольняє умовам, не існує, програма повинна вивести одне число **0**. Якщо знайдеться декілька маршрутів, то вивести той що починається ближче до точки старту експедиції.

**Приклади**

|  |  |
| --- | --- |
| **Введення** | **Виведення** |
| 7  18 10 15 20 20 10 3 | 3 6 |
| 3  9 8 5 | 0 |

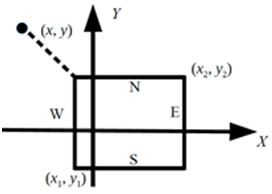
**Задача Lattice** Є прямокутник розміром M\*N, що складається з клітинок 1\*1. Знайдіть кількість квадратів, всі вершини яких є вершинами клітинок. Сторони квадратів НЕ обов’язково паралельні до сторін прямокутника.

**Технічні умови.** Програма  **Lattice** читає з пристрою стандартного введення два цілих числа - розміри прямокутника **M** та **N*(1≤M,N≤10000)*** і виводить на пристрій стандартного виведення шукану кількість квадратів.

**Приклад**

***Введення 2 3***

***Виведення 10***

**Задача Raft** Петрик влітку відпочиває у бабусі в селі. Особливо йому подобається купатись на сільському озері. Посередині озера плаває пліт, який має форму прямокутника. Сторони плота спрямовані уздовж паралелей і меридіанів. Введемо систему координат, в якій вісь ОХ направлена на схід, а вісь ОY - на північ. Нехай південно-західний кут плоту має координати ***(x1, y1)***, північно-східний кут - координати ***(x2, y2)***. Петрик знаходиться в точці з координатами ***(x, y)***. Визначте, до якої сторони плоту (північної, південної, західної чи східної) або до будь-якого кута плоту (північно-західному, північно-східному, південно-західному, південно-східному) Петрику потрібно плисти, щоб якомога швидше дістатися до плоту.

***Технічні умови. Програма Raft***читає з пристрою стандартного введення шість чисел в наступному порядку: ***x1, y1*** (координати південно-західного кута плоту), ***x2, y2*** (координати північно-східного кута плоту), ***x, y*** (координати Петрика). Всі числа цілі і по модулю не перевершують 100. Гарантується, що ***x1 < x2, y1 < y2, x ≠ x1, x ≠ x2, y ≠ y1, y ≠ y2***, координати Петрика знаходяться поза плотом.  Якщо Петрику слід пливти до північної сторони плоту, програма повинна вивести на пристрій стандартного виведення символ «N», до південної - символ «S», до західної - символ «W», до східної - символ «E». Якщо Петрику слід пливти до кута плоту, потрібно вивести один з наступних рядків: «NW», «NE», «SW», «SE».

***Приклад***

|  |  |
| --- | --- |
| **Введення** | **Виведення** |
| -1 -2  5 3 -4   6 | NW |

***Задача Balloon***. Герої Жюля Верна мандрували на повітряній кулі. У них був таймер та висотомір. Таймер відміряв час з моменту старту в годинах, а висотомір – висоту, на якій у цю мить знаходиться куля.  Результати вимірів вони  записували,  інколи – не щогодини, але якщо  в час ***T1*** висота складала ***Y1***, а в час ***T2*** висота складала ***Y2***, то між годинами ***T1*** та ***T2*** висота рівномірно змінювалася від ***Y1*** до ***Y2***. Мандрівники вирішили дізнатися, скільки часу тривав самий затяжний підйом (тобто максимальний проміжок часу, на якому висота зростала).

**Технічні умови**. Програма ***Balloon*** читає з пристрою стандартного введення (клавіатури) число ***N (1 ≤ N ≤105)*** – кількість найдених записів. В наступних ***N***стрічках по 2 цілих числа ***А*** та ***В***, де ***А***– номер години польоту, а ***В***– висота в цей момент часу.***(1 ≤*** ***А, В ≤*** ***109).*** Гарантується, що всі ***А*** різні. Програма виводить на пристрій стандартного введення (екран) єдине число – тривалість  самого затяжного підйому у годинах.

***Приклад***

***Введення  Виведення***

**10              3**

**1 6**

**2 20**

**3 15**

**4 10**

**6 13**

**7 20**

**8 20**

**9 20**

**10 20**

**11 21**